

Lab 2_1 보고서

20210479 이주현

1 개요

Lab 2_1에서는 Bool algebra를 단순화하는 방법인 Karnaugh-Map(이하 K-Map)을 학습하고, 단순화 이전의 expression을 구현해 보는 것을 목표로 한다.

2 이론적 배경

(1) Bool algebra의 단순화

Bool algebra는 그것이 실제로 구현되었을 때 사용된 와이어와 논리 게이트의 개수로 복잡도를 평가한다. 단순화란 Bool algebra의 복잡도를 낮추는 과정으로, 즉 구현에 사용하는 와이어와 논리 게이트의 개수를 줄이는 과정이라고 할 수 있다. 이를 통해 소비 전력을 줄이고, 작동 속도를 높일 수 있다는 장점을 갖는다.

(2) Karnaugh-Map (K-Map)

대표적인 Bool algebra의 단순화 방법으로, Bool algebra를 minimal sum of products form으로 나타낼 수 있는 graphical technique이다. Map에 표시된 box를 2^n 개씩 묶어서 n 개의 literal을 없애는 방식으로 단순화할 수 있다.

(2) 2-Bit Magnitude Comparator

2-Bit Magnitude Comparator는 서로 다른 두 개의 2-Bit binary number가 입력으로 주어졌을 때, 둘의 대소 관계를 판별하여 진리값을 출력하는 회로를 의미한다.

3 실험 준비

(1) “A > B”에 대한 진리표 작성 및 식 표현

입력		A1A0			
		00	01	11	10
B1 B0	00	0	1	1	1
	01	0	0	1	1
	11	0	0	0	0
	10	0	0	1	0

따라서 이를 식으로 표현하면 $A1'A0B1'B0' + A1A0B1'B0' + A1A0'B1'B0' + A1A0B1'B0 + A1A0'B1'B0 + A1A0B1B0'$ 이다.

(2) “A = B”에 대한 진리표 작성 및 식 표현

입력		A1A0			
		00	01	11	10
B1 B0	00	1	0	0	0
	01	0	1	0	0
	11	0	0	1	0
	10	0	0	0	1

따라서 이를 식으로 표현하면 $A1'A0'B1'B0' + A1'A0B1'B0 + A1A0B1B0 + A1A0'B1B0'$ 이다.

(3) “A < B”에 대한 진리표 작성 및 식 표현

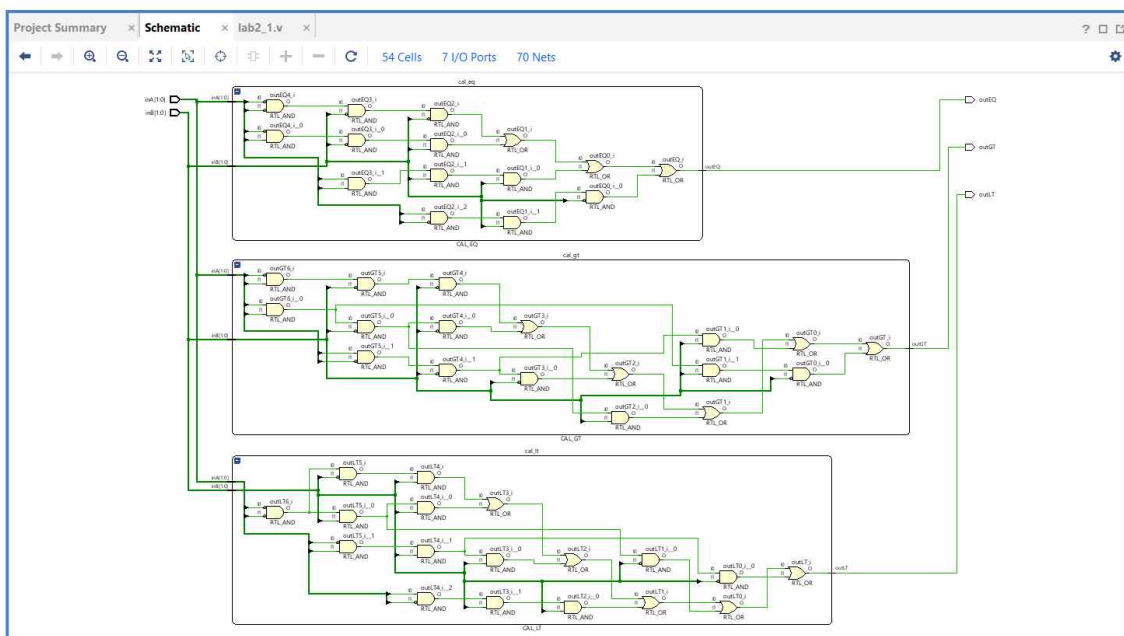
입력		A1A0			
		00	01	11	10
B1	00	0	0	0	0
	01	1	0	0	0
B0	11	1	1	0	1
	10	1	1	0	0

따라서 이를 식으로 표현하면 $A1'A0'B1'B0 + A1'A0'B1B0 + A1'A0B1B0 + A1A0'B1B0 + A1'A0'B1B0' + A1'A0B1B0'$ 이다.

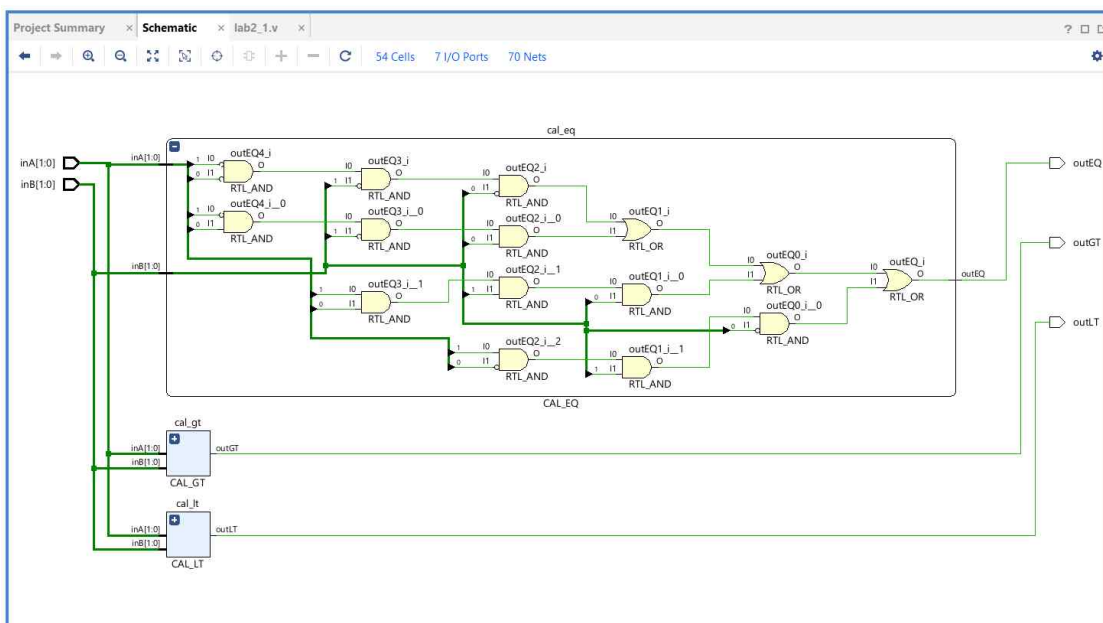
4 결과

작성한 회로도(RTL analyzer schematic)는 다음과 같았다. 회로도 미루어 보아 구현이 제대로 된 것을 알 수 있다.

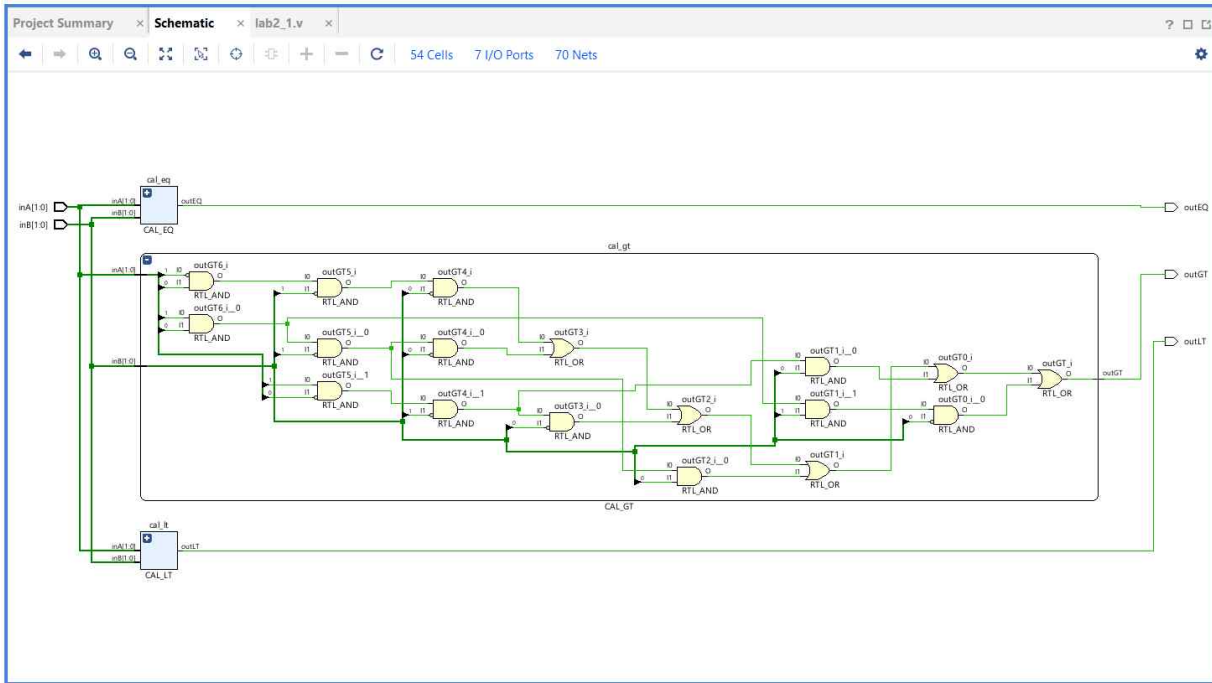
(1) 전체 회로도



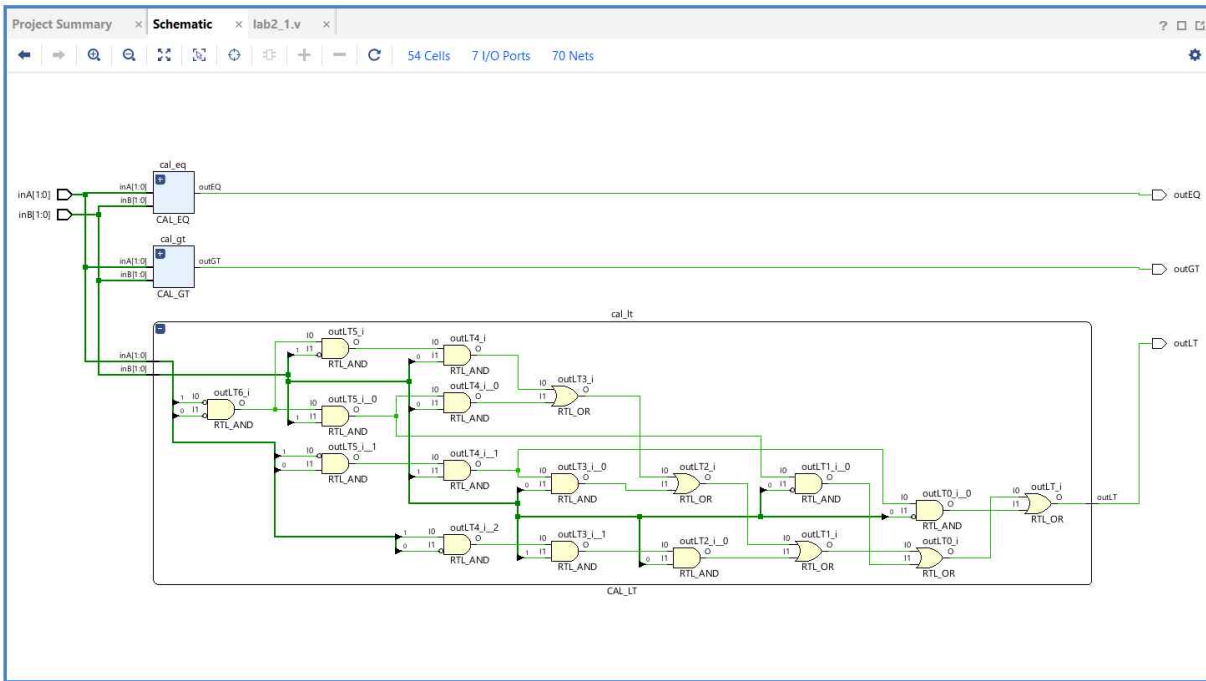
(2) EQ 확대



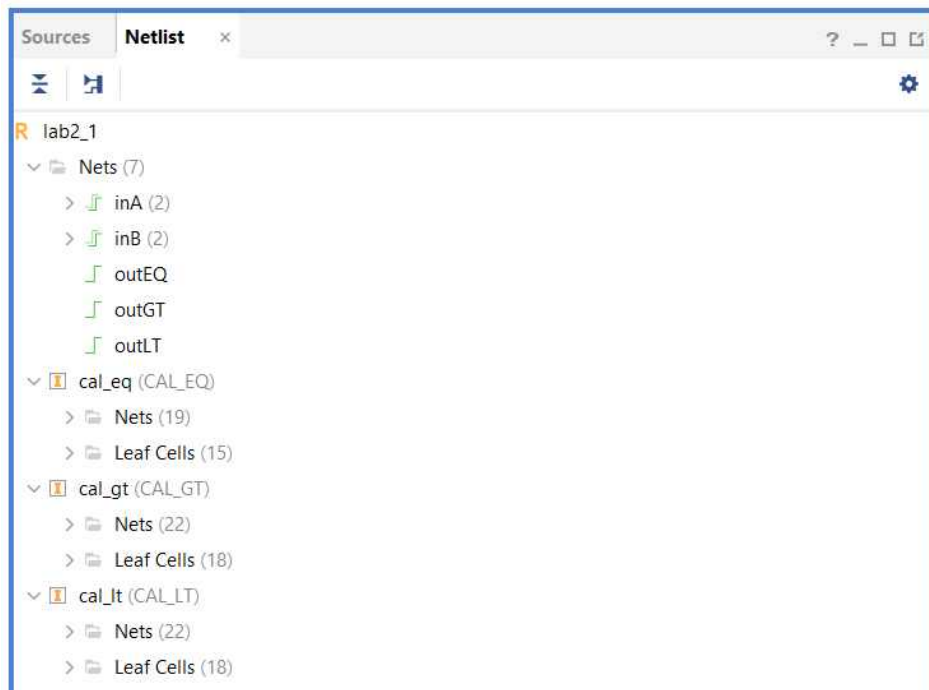
(3) GT 확대



(4) LT 확대



(5) Netlist



5 논의

wire array의 index가 $\text{inA}[1] \rightarrow \text{inA}[0]$ 순서인 것이 다른 언어와 달라 코딩을 하면서 살짝 헷갈린다는 느낌을 받았다. 이때 미리 써놓은 진리표와 Bool algebra expression을 보며 도움을 받을 수 있었다.

Lab 2_2 보고서

20210479 이주현

1 개요

Lab 2_2에서는 Bool algebra를 단순화하는 방법인 Karnaugh-Map(이하 K-Map)을 학습하고, 단순화된 식을 구현하고 전후를 비교하여 단순화의 효과를 확인하는 것을 목표로 한다.

2 이론적 배경

(1) Bool algebra의 단순화

Bool algebra는 그것이 실제로 구현되었을 때 사용된 와이어와 논리 게이트의 개수로 복잡도를 평가한다. 단순화란 Bool algebra의 복잡도를 낮추는 과정으로, 즉 구현에 사용하는 와이어와 논리 게이트의 개수를 줄이는 과정이라고 할 수 있다. 이를 통해 소비 전력을 줄이고, 작동 속도를 높일 수 있다는 장점을 갖는다.

(2) Karnaugh-Map (K-Map)

대표적인 Bool algebra의 단순화 방법으로, Bool algebra를 minimal sum of products form으로 나타낼 수 있는 graphical technique이다. Map에 표시된 box를 2^n 개씩 묶어서 n 개의 literal을 없애는 방식으로 단순화할 수 있다.

(2) 2-Bit Magnitude Comparator

2-Bit Magnitude Comparator는 서로 다른 두 개의 2-Bit binary number가 입력으로 주어졌을 때, 둘의 대소 관계를 판별하여 진리값을 출력하는 회로를 의미한다.

3 실험 준비

(1) “A > B”에 대한 K-Map 작성 및 식 단순화

입력		A1A0			
		00	01	11	10
B1 B0	00	0	1	1	1
	01	0	0	1	1
	11	0	0	0	0
	10	0	0	1	0

따라서 단순화된 식은 $A1B1' + A0B1'B0' + A1A0B0'$ 이다.

(2) “A = B”에 대한 K-Map 작성 및 식 단순화

입력		A1A0			
		00	01	11	10
B1 B0	00	1	0	0	0
	01	0	1	0	0
	11	0	0	1	0
	10	0	0	0	1

따라서 단순화된 식은 $A1'A0'B1'B0' + A1'A0B1'B0 + A1A0B1B0 + A1A0'B1B0'$ 이다.

(3) “A < B”에 대한 K-Map 작성 및 식 단순화

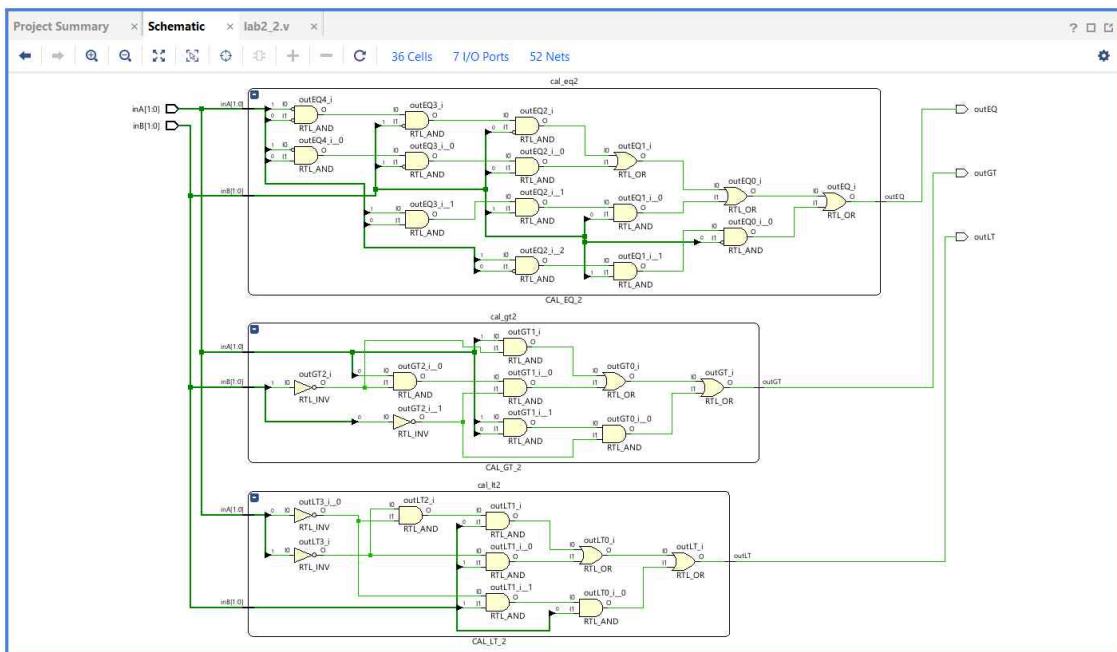
입력		A1A0			
		00	01	11	10
B1	00	0	0	0	0
	01	1	0	0	0
B0	11	1	1	0	1
	10	1	1	0	0

따라서 단순화된 식은 $A1'A0'B0 + A1'B1 + A0'B1B0$ 이다.

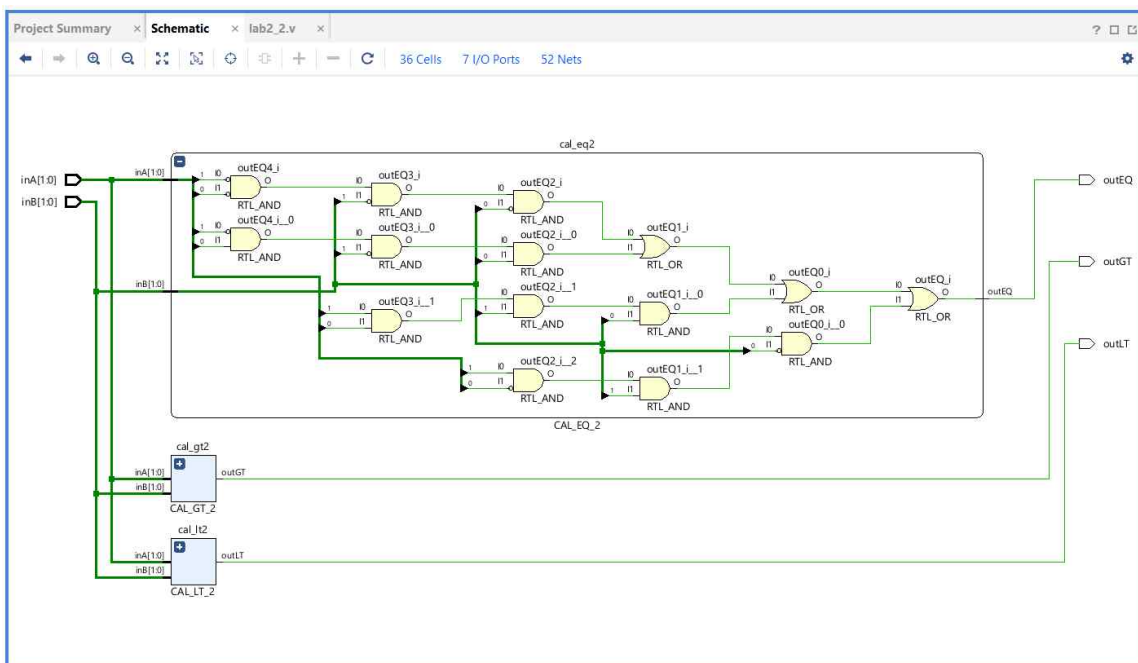
4 결과

작성한 회로도(RTL analyzer schematic)는 다음과 같았다. 회로도 미루어 보아 구현이 제대로 된 것을 알 수 있다.

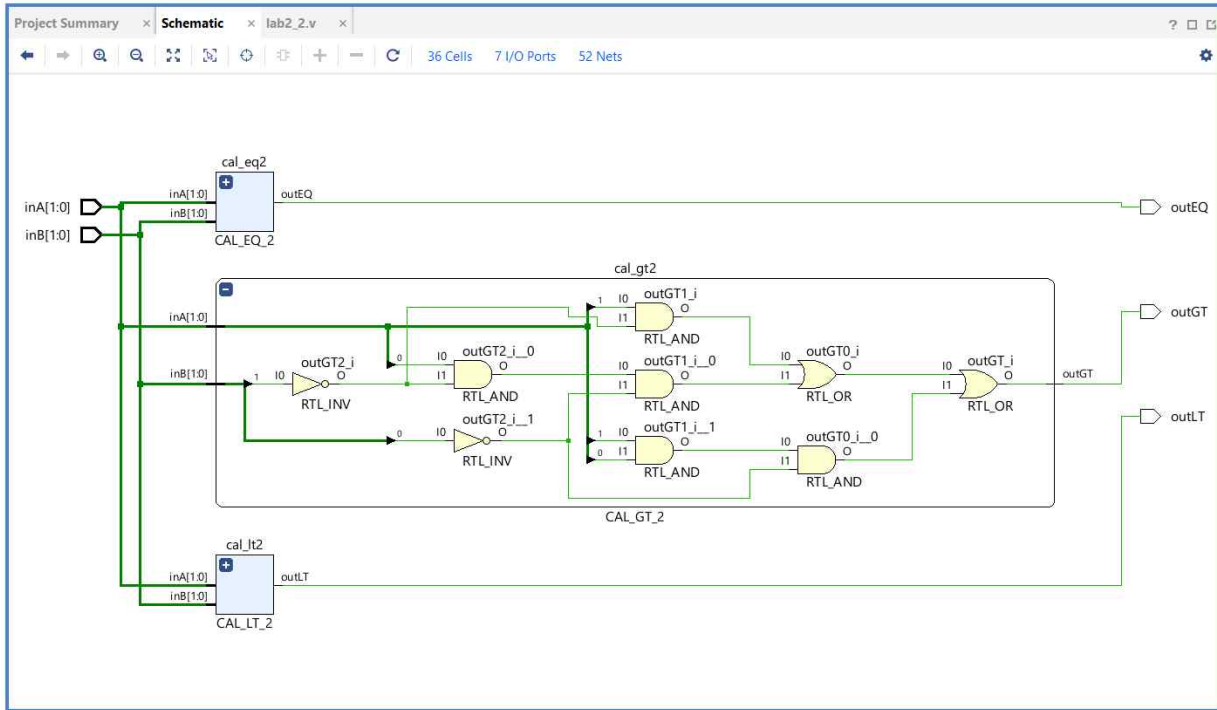
(1) 전체 회로도



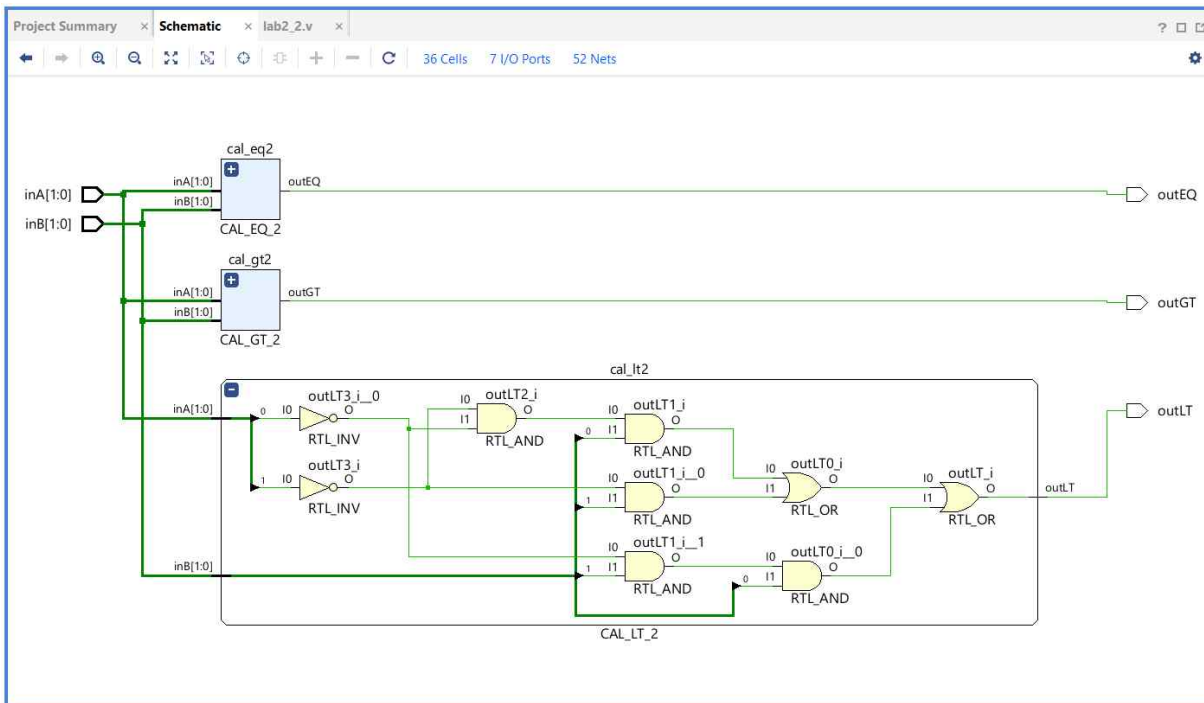
(2) EQ 확대



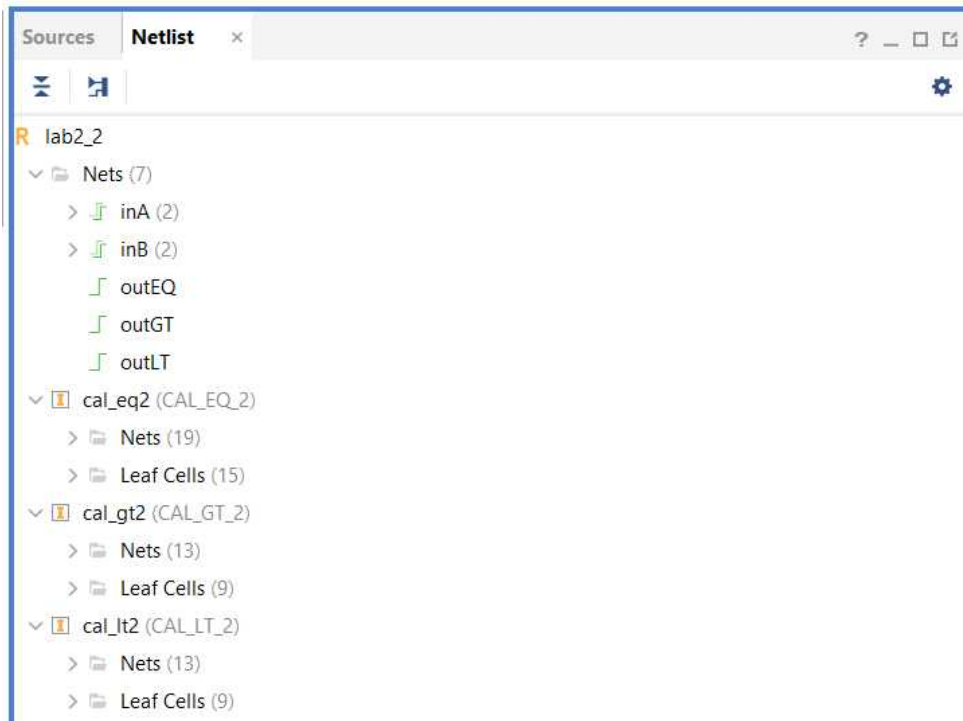
(3) GT 확대



(4) LT 확대



(5) Netlist



Lab2_1과 Lab2_2의 Netlist를 비교해 보면, 단순화를 하였을 때 와이어와 논리 게이트의 개수가 줄어든 것을 볼 수 있다. (단, EQ의 경우 단순화 전과 후의 식이 동일하므로 와이어와 논리 게이트의 개수에 변화가 없다.) 즉, Boolean algebra의 단순화 과정을 통해 작동 속도를 높이는 효과를 가져올 수 있음을 확인할 수 있다.

5 논의

4-variable K-Map에서 인접한 box를 묶을 수 있는지 판단하는 과정에서 어려움을 겪었다. Ch 02에 나온 개념을 다시 한번 복습하고, 예제를 다시 풀어보면서 이를 극복할 수 있었다.